

Application of Pretrained Deep Neural Networks to Large Vocabulary Speech Recognition

Navdeep Jaitly¹, Patrick Nguyen², Andrew Senior², Vincent Vanhoucke²

¹Department of Computer Science, University of Toronto

²Google Inc.



Abstract

The use of Deep Belief Networks (DBN) to pretrain Deep Neural Networks (DNN) has recently led to a resurgence in the use of Artificial Neural Network - Hidden Markov Model (ANN/HMM) hybrid systems for Automatic Speech Recognition (ASR). In this paper we report results of a DBN-pretrained context-dependent 'DNN/HMM' system trained on two datasets that are much larger than any reported previously with DBN-pretrained ANN/HMM systems - 5870 hours of Voice Search and 1400 hours of YouTube data. On the first dataset, the pretrained ANN/HMM system outperforms the best Gaussian Mixture Model - Hidden Markov Model (GMM/HMM) baseline, built with a much larger dataset by 3.7% absolute WER, while on the second dataset, it outperforms the GMM/HMM baseline by 4.7% absolute. Maximum Mutual Information (MMI) fine tuning and model combination using Segmental Conditional Random Fields (SCARF) give additional gains of 0.1% and 0.4% on the first dataset and 0.5% and 0.9% absolute on the second dataset.

Introduction

Recent advances in Machine Learning have led to the development of algorithms which can be used to train deep models. One of these approaches is the Deep Belief Network (DBN), a multi-layered generative model which can be trained greedily, layer by layer, using a model known as a Restricted Boltzmann Machine at each layer [1]. It has been empirically observed that using the parameters of a Deep Belief Network to initialize (a.k.a "pretrain") a deep neural network before fine tuning with backpropagation leads to improved performance of the deep neural network on discriminative tasks. The successful training of deep neural networks (DNN) on several tasks (with or without pretraining) has led to its widespread adoption in speech recognition systems where DNN/HMM hybrid systems have demonstrated tremendous gains [2, 4, 5, 3].

In this paper we report our results on experiments with DNN/HMM hybrids on Google's datasets and language models that are much larger than datasets and language models previously reported in the literature in this area.

Datasets and Baselines

Voice Search The training data for the Voice Search system consisted of approximately 5780 hours of data from mobile Voice Search and Android Voice Input. The baseline model used was a triphone HMM with decision-tree clustered states. The acoustic data was contiguous frames of PLP features that were transformed by Linear Discriminant Analysis (LDA). Semi-Tied Covariances (STC) were used in the GMMs to model the LDA transformed features. Boosted-MMI was used to train the model discriminatively. This generated a CD model with 7969 states.

You-tube The training data for the YouTube system consisted of approximately 1400 hours of data from YouTube. The system used 9-frame MFCCs that were transformed by LDA and SAT was performed. Decision-tree clustering was used to obtain 17552 triphone states, and STCs were used in the GMMs to model the features. The acoustic models were further improved with BMMI. During decoding, Constrained Maximum Likelihood Linear Regression (CMLLR) and Maximum Likelihood Linear Regression (MLLR) transforms were applied.

Summary of Baselines

Name	# of hours	CMLLR?	WER
Voice Search	>>6K	No	16.0
YouTube	>>1400	Yes	52.3

Table 1: Baselines used for study.

Overview

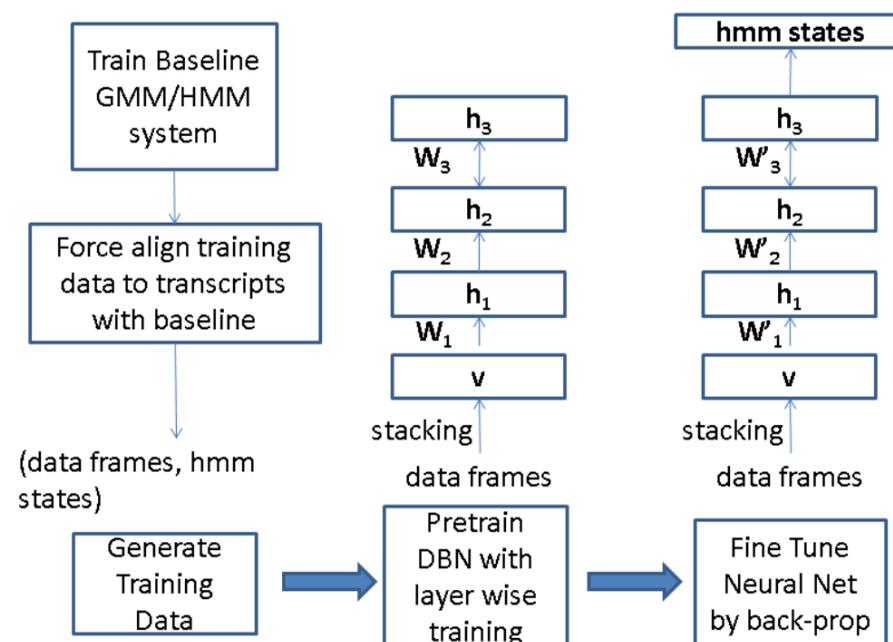


Figure 1. Pipeline for training ANN/HMM hybrid system

The ANN/HMM hybrid models were trained in three stages as shown in figure 1. First, a baseline GMM/HMM system was trained and forced alignment was used to associate each frame of data with a target HMM state. Then, a DBN was trained on the acoustic data (which may be MFCC vectors, log filterbanks, or speaker adapted features stacked together) and the weights of the DBN were used to initialize a neural network, which was then trained to predict the HMM state from the acoustic data, using back-propagation.

Further discriminative training of the learnt neural network was then performed using MMI. Lastly, SCARF was used for model combination of DNN/HMM results with the GMM/HMM system.

Results

Name	Model	WER(%)
Voice Search	GMM-HMM baseline	16.0
	DBN pretrained ANN/HMM with sparsity	12.3
	+ MMI	12.2
	+ system combination with SCARF	11.8
YouTube	GMM-HMM baseline	52.3
	DBN pretrained ANN/HMM with sparsity	47.6
	+ MMI	47.1
	+ system combination with SCARF	46.2

Table 2: Summary of Results

Methods and Experiments

Neural Network Architecture Based on exploratory experiments with the Broadcast news database, we chose to use four hidden layers with 2560 units per layers as the architecture of choice for Voice Search. For You-tube we also used a neural network with 4 hidden layers. However, we chose to use 1000 units at all layers but the lowest layer (where we used 2000 units), because of computational considerations - the targets had a very high output dimensionality of 17552.

Neural Network Training The models were trained on a dual CPU Intel Xeon DP Quad Core E5640 machine with Ubuntu OS equipped with four NVIDIA Tesla C2070 Graphics Processing Units. Each job was performed on a single CPU with a single GPU board. Data were loaded on to CPU memory in big mini-batches of 20 hours for Voice Search, and 17.5 hours for YouTube. These were then loaded into the GPU, and randomly permuted. Mini-batches of size 200 for Voice Search and 500 for YouTube were built by cycling through these permuted vectors. Model parameters were all kept and updated on GPU memory itself. Average gradients were computed on the mini-batches and parameters were updated with a learning rate of .04 for the top two layers of the network and 0.02 for the others, with a momentum of 0.9. Each DBN layer was pre-trained for one epoch as an RBM and then the resulting ANN was discriminatively fine-tuned for one epoch. Weights with magnitudes below a threshold were then permanently set to zero before a further quarter epoch of training. All the computations involved in training the DBN (matrix multiplications, sampling etc) and the Neural Network (matrix multiplications, etc) were performed on the GPU using the Cudamat library [8].

Discriminative training with MMI Discriminative training of the DNN/HMM model was performed using a gradient update rule similar to that described in [6]. Gradient descent learning with momentum was performed over large mini-batches of size equal to $1/20^{th}$ of the entire training data set.

Model Combination with GMM/HMMs using SCARF SCARF was used to combine results from the GMM/HMM model with the results from the DNN/HMM model[7].

References

- [1] G.E. Hinton and S. Osindero and Y. Teh. A Fast Learning Algorithm for Deep Belief Nets. *Neural computation* (18), 1527-54, 2006
- [2] G.E. Dahl and D. Yu and L. Deng and A. Acero. Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition. *IEEE Trans. Audio, Speech, and Language Processing* (99), 2012.
- [3] F. Seide and G. Li and D. Yu. Conversational Speech Transcription Using Context-Dependent Deep Neural Networks. *Inter-speech*, 437-440, 2012.
- [4] A. Mohamed and G.E. Dahl and G.E. Hinton. Acoustic Modeling using Deep Belief Networks. *IEEE Trans. Audio, Speech, and Language Processing* (99), 2012.
- [5] A. Mohamed and T.N. Sainath and G.E. Dahl and B. Ramabhadran and G.E. Hinton and M.A. Picheny. Deep Belief Networks using Discriminative Features for Phone Recognition *Proceedings, ICASSP*, 5060-5063, 2012
- [6] B. Kingsbury. Lattice-based Optimization of Sequence Classification Criteria for Neural-Network Acoustic Modeling. *Proceedings, ICASSP*, 3761-3764, 2009
- [7] G. Zweig, P. Nguyen et. al. Speech Recognition with Segmental Conditional Random Fields: A summary of the JHU CLSP 2010 Summer Workshop. *Proceedings, ICASSP*, 5044-5047, 2011
- [8] V. Mnih. Cudamat: a CUDA-based Matrix Class for Python. *Technica Report 004* University of Toronto, 2009